

# Efficiency Testing of Active Noise Control by Acoustic Field Modeling

Attila Szarvas<sup>1</sup>, László Sujbert<sup>1\*</sup>

Received 31 July 2015; accepted 24 October 2015

## Abstract

The efficiency of active noise control (ANC) is largely dependent on the arrangement of the microphones and loudspeakers, and the acoustic environment as well. The paper presents a simulation tool, which allows the performance evaluation of ANC systems without the difficulty of physical experiments. The proposed simulation integrates sound field calculation and noise control. Field calculation is based on robust FDTD techniques, while the discrete time controller is realized in a usual way. The simulation is performed in three stages. First the program determines the discrete transfer function models of the acoustic paths between specific points of interest. In the next stage the controller's operation is evaluated without field simulation. Having the steady state parameters of the adaptive controller, the field simulation is re-run for the reverberation time providing the sound pressure map. Sound field calculation is implemented on modern GPGPU devices. A series of ANC experiments were compared to the results of the simulations, and they prove the applicability of our simulation method in a convincing manner.

## Keywords

active noise control, acoustic field modeling, simulation

## 1 Introduction

Active noise control (ANC) relies on generating the inverse of unwanted noises, and arranging a scenario where destructive interference results in their elimination [1,2]. ANC has a thoroughly established theoretical background and there is a large number of practical solutions based on the idea. The difficulty of a particular case however i.e. the efficiency by which we can eliminate the unwanted signals is largely dependent on the details of the noise control scenario [2,3].

Our investigations concern particularly ANC systems operating under room acoustic conditions. In these situations the multiple successive reflections of wavefronts result in transfer functions of high complexity. The medium of actuation is the acoustic field, which is usually shared by the noise sources, actuators, error and reference microphones alike. This creates close coupling between the variables of the noise canceling problem and reduces the orthogonality of signals. One way to improve the performance of ANC in suppressing stochastic disturbances is using control systems relying on a large number of inputs, yielding a lot of information about the acoustic field. Such noise canceling setups have a very large degree of freedom, and finding the optimal parameters requires a lot of experimentation [4,5,6].

It has been concluded that a software tool, which would allow us to simulate the operation of such ANC systems, would alleviate much of the difficulty in physically arranging the experiments. Besides allowing us to iterate faster with our tests this tool would also deepen our understanding of ANC system behavior under such conditions.

Our criteria for the proposed tool, is that it is capable of the joint simulation of ANC systems and physical wave propagation in room acoustic environments. While there are proprietary industry grade room modelling solutions for auralization purposes [7,8,9], they don't offer the level of interactivity that we require. Consequently we have decided to create the simulation software presented herein tailored specifically to our use-cases.

The first of the two major components of our simulation shall be the acoustic field modeler. This gives a numerical solution of the acoustic field values in the time domain and it allows

<sup>1</sup> Department of Measurement and Information Systems,  
Faculty of Electrical Engineering and Informatics,  
Budapest University of Technology and Economics,  
H-1521 Budapest, P.O.B. 91, Hungary

\* Corresponding author, e-mail: [sujbert@mit.bme.hu](mailto:sujbert@mit.bme.hu)

us to observe the behavior of the noise cancelling module as if it was operating in practice. After careful consideration we have chosen the iterative solution of the acoustic wave equation by the *finite-difference time domain* (FDTD) method [10]. The competing *finite-element* (FEM) [11] and *boundary-element methods* (BEM) [12] are more complicated, and their ability to handle complex surface geometries is not needed in our experiments. The implementation of the FDTD method is based on the thorough work of Kowalczyk [13]. It has already been referred to by other authors dealing with room acoustics [14,15].

The other major component shall implement the noise controller. For our purposes we have chosen the *filtered-error LMS* (ELMS) method, belonging to the *least mean square* (LMS) based controller family [16]. The ELMS algorithm utilizes an adaptive FIR filter, which is inherently stable, thus preferable to IIR methods. We have rejected the competing *filtered-reference LMS* (XLMS) methods, because in the case of large number of reference signals, filtering those instead of the few error signals would carry a much higher computational cost [16].

Our simulation tool integrates these two above mentioned components. This modular structure allows us to modify ANC signals, variables and even system structure independently from the acoustic parameters.

This paper is structured as follows. In Section 2 we first introduce the signal processing problem of acoustic noise cancelling. Then the physical model of wave propagation is discussed in Section 3. Section 4 describes the implementation details of the physical modeling, while Section 5 discusses its technological implications. Section 6 presents the simulation tasks, and results provided by our tool. Finally, Section 7 concludes our findings.

## 2 Active noise control

### 2.1 Statement of the acoustic noise canceling problem

Our investigations focus on a general class of noise control systems that are capable of effectively damping stochastic disturbances such as naturally occurring wideband signals. Future values of such signals are uncorrelated with the present and past measurements, and thus we cannot predict them. Consequently the noise has to be continuously sampled using *reference microphones* providing the *reference signals*. There are *error microphones* in the cancelation area providing the *error signal* inputs for the controller [2,4]. The control systems pertaining to our studies assume a linear media i.e. changes in the observed quantities are a linear function of the actuation. This is applicable for acoustic fields given sound pressure levels occurring in practice. Expanding on the example shown by Fig. 1a and 1b we describe a problem with multiple noise sources, reference and error microphones. Figure 1a shows the wave propagation from the noise sources to the microphones, while Fig. 1b depicts the control signal paths. The number of

noise sources, error and reference microphones, and actuator loudspeakers can be chosen independently of each other.

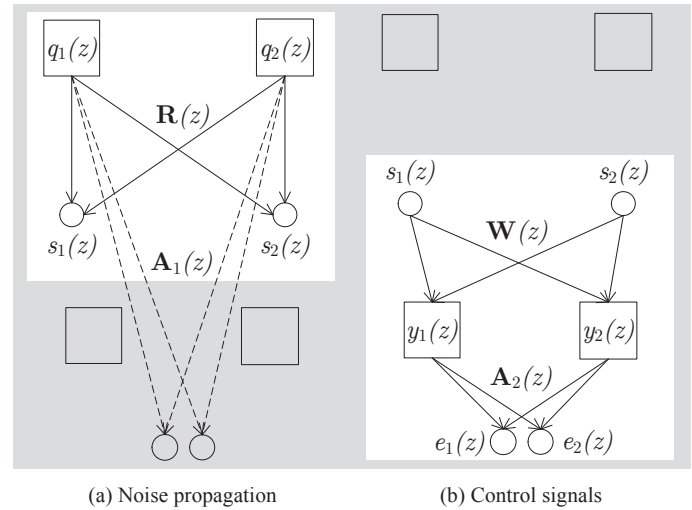


Fig. 1 Transfer paths in an ANC problem

First let us consider Fig. 1a. The primary sources are represented by the signals  $q_1(z)$  and  $q_2(z)$ . Their signal propagates to the reference microphones through the path  $\mathbf{R}(z)$  yielding the signals  $s_1(z)$  and  $s_2(z)$  at the reference microphones. The noise reaches the error microphones through  $\mathbf{A}_1(z)$ , which is the primary path. In this model point sources are used. Distributed noise sources can be represented by multiple point sources, if necessary.

Figure 1b shows the control signals and the corresponding paths. The reference signals  $s_1(z)$  and  $s_2(z)$  are filtered by the controller  $\mathbf{W}(z)$  producing the secondary noise sources  $y_1(z)$  and  $y_2(z)$ . This noise reaches the error microphones through  $\mathbf{A}_2(z)$ , which is the secondary path. The signals at the error microphones are  $e_1(z)$  and  $e_2(z)$  resulting from the superposition of the primary and secondary waves. All the signals and transfer functions shown in Fig. 1a and 1b are modeled as discrete ones, and are expressed in the  $z$ -domain.

In ideal case, the secondary noise completely cancels the primary one, i.e. the error signal is zero [6,17]:

$$\mathbf{e}(z) = [\mathbf{A}_2(z) \cdot \mathbf{W}(z) \cdot \mathbf{R}(z) - \mathbf{A}_1(z)] \mathbf{q}(z) = 0, \quad (1)$$

where  $\mathbf{e}(z)$  and  $\mathbf{q}(z)$  are the vectors formed by the error and primary source signals, respectively. Expressing the controller transfer function  $\mathbf{W}(z)$  we get:

$$\mathbf{W}(z) = \mathbf{A}_2^{-1}(z) \cdot \mathbf{A}_1(z) \cdot \mathbf{R}^{-1}(z). \quad (2)$$

If (2) holds, the error signal is really zero. However, matrices  $\mathbf{A}_2(z)$  and  $\mathbf{R}(z)$  are usually not invertible, as there are different number of primary and secondary sources, reference and error microphones. Hence, instead of the ordinary inverse the pseudoinverse is to be used, which guarantees only the minimization of the error signal in the least mean squares sense:

$$\mathbf{W}(z) = \mathbf{A}_2^\#(z) \cdot \mathbf{A}_1(z) \cdot \mathbf{R}^\#(z), \quad (3)$$

where the hash mark stands for pseudoinverse [6]. The implementation of the optimal controller transfer function  $\mathbf{W}(z)$  is also problematic. In general the resulting  $\mathbf{W}(z)$  has an *infinite impulse response*, furthermore, if  $\mathbf{A}_2(z)$  and  $\mathbf{R}(z)$  are not of minimum phase, their inverses are unstable. Widely used adaptive controllers are based on the LMS algorithm. In such controllers  $\mathbf{W}(z)$  is modeled by a *finite impulse response* filter. If it is sufficiently long, a fair approximation can be achieved.

Another problem is that the adaptive filter can only realize a causal  $\mathbf{W}(z)$  [17], and its approximation ability strongly depends on the relation between  $\mathbf{A}_2(z)$ ,  $\mathbf{R}(z)$ , and  $\mathbf{A}_1(z)$ . Acoustic transfer functions may include significant delays, as wave propagation time is much longer than the sampling period. Considering again Eq. (3), the inverses contribute to a *negative delay* in  $\mathbf{W}(z)$ , while the primary path has a regular, *positive delay*. Stable approximation of the unstable inverses results in a requirement of *negative delay* in  $\mathbf{W}(z)$  again. Hence, if the primary path  $\mathbf{A}_1(z)$  has a sufficient delay, which overcomes those of  $\mathbf{A}_2(z)$  and  $\mathbf{R}(z)$ , a causal  $\mathbf{W}(z)$  may exist, yielding good approximation, thus resulting in fair noise suppression.

## 2.2 Noise controller

Figures 1a and 1b have shown the wave propagation paths rather than the controller. As it utilizes reference signals, the controller is of a feedforward type [4,5]. The controller implemented in our simulation framework is based on an extension of the least mean square (LMS) algorithm [16]. This is the most common choice since its iterative behavior allows the system to be constantly adapted to the observed acoustic transfer relations, and it can match slow changes in the environment while keeping the error small. In a static situation it ensures that the power of the remainder noise is minimized. The two most widely used variants of the LMS algorithm for noise cancelling purposes are the filtered reference or filtered-X LMS (XLMS) and filtered error LMS (ELMS) algorithms. The difference between the algorithms is how they utilize the information about the secondary paths. The former filters the reference signals with the approximated transfer function of the secondary path, and the latter filters the error signals with the approximated delayed inverse of the secondary path [16].

The ELMS algorithm presented by [16] shows faster convergence and is much more suitable for systems with many reference signals regarding both the computational and memory requirements. For this reason only the ELMS algorithm is implemented in the simulation software and will be introduced in a detailed fashion.

Figure 2 shows the signal processing block diagram of the ELMS algorithm. The signals introduced in Fig. 1a and 1b can be recognized here. Note that all the signals are expressed in the discrete time domain, as it is more convenient for the

algorithm description. The so-called desired signal  $\mathbf{d}_n$  is the primary noise at the error microphones. The reference signal  $\mathbf{s}_n$  is filtered by the controller  $\mathbf{W}(z)$  producing the secondary noise  $\mathbf{y}_n$ , and the remaining error signals are denoted with  $\mathbf{e}_n$ . All the signals are vectors and expressed at the time instant  $n$ . The controller  $\mathbf{W}_n$  in the discrete time domain is a matrix of transversal filters each element of which is updated by the ELMS rule. Each filter comprises  $N$  coefficients. The  $\hat{\mathbf{A}}_2^\#(z)$  filter is the estimated delayed inverse of the secondary path. As  $\mathbf{A}_2(z)$  is not of minimum phase, it has no stable inverse. However, its delayed version can be sufficiently approximated [16]. It is constructed in the system's identification mode prior to noise canceling. It is estimated with the simple LMS algorithm such that

$$\hat{\mathbf{A}}_2^\#(z) \mathbf{A}_2(z) \cong z^{-D} \mathbf{I} \quad (4)$$

The hash mark stands again for the pseudoinverse, since the number of reference microphones is not necessarily the same as the number of error microphones. The value of the delay  $D$  is chosen by practical considerations and the correctness of its value has to be experimentally verified.

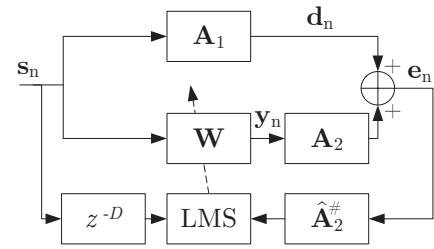


Fig. 2 Block diagram of the ELMS algorithm

The update step of the adaptive filter is given by the following equation:

$$\mathbf{W}_{i,n+1} = \mathbf{W}_{i,n} + \mu \mathbf{r}_n \mathbf{s}_{n-D-i}^T, \quad i = 0 \dots N-1, \quad (5)$$

where  $\mathbf{W}_{i,n}$  denotes a matrix composed of the  $i$ th coefficients of the controller filter matrix,  $\mathbf{s}_{n-D-i}^T$  stands for the reference signal vector delayed by  $D+i$  samples, and  $\mu$  is an experimentally chosen step coefficient. The vector  $\mathbf{r}_n$  is the filtered error signal, defined as it follows:

$$\mathbf{r}(z) = \hat{\mathbf{A}}_2^\#(z) \mathbf{e}(z). \quad (6)$$

The filter matrix  $\hat{\mathbf{A}}_2^\#(z)$  is a model defined by (4), the elements of which are transversal filters of the same length as those of  $\mathbf{A}_2(z)$ .

Figure 3 shows the identification process for  $\hat{\mathbf{A}}_2^\#(z)$  for the simple two reference, two error microphone case, where the ordinary inverse  $\hat{\mathbf{A}}_2^{-1}(z)$  can be determined. During this phase the ANC system generates white noise and sends it to the actuators. It needs to generate as many uncorrelated noise signals as the number of actuators. The signals generated by the system

are labeled  $s_{1,n}$  and  $s_{2,n}$  on the block diagram. The acoustic system's transfer function is represented by the  $A_2(z)$  block. The signals of the error microphones are labeled  $d_{1,n}$  and  $d_{2,n}$ . These signals are then filtered by  $\hat{A}_2^{-1}(z)$ , and the results are compared with the delayed  $s_{1,n}$  and  $s_{2,n}$  identification signals.  $D$  is the delay determined by practical considerations and should be significantly greater than the delay of  $A_2(z)$ . The error microphone signals and the estimation error is used to adapt  $\hat{A}_2^{-1}(z)$ . Now  $\hat{A}_2^{-1}(z)$  is also expressed as  $\hat{A}_2^{-1}(z)$  in the discrete time domain. It is a matrix of transversal filters of  $M$  coefficients whose elements are updated by the LMS rule:

$$\hat{A}_{2,i,n+1}^{-1} = \hat{A}_{2,i,n}^{-1} + \mu \mathbf{e}_n \mathbf{d}_{n-i}^T, \quad i = 0 \dots M-1, \quad (7)$$

where  $\hat{A}_{2,i,n}^{-1}$  denotes a matrix composed of the  $i$ th coefficients of the inverse filter matrix,  $\mathbf{d}_{n-i}^T$  stands for the microphone signal vector delayed by  $i$  samples, and  $\mu$  is an experimentally chosen step coefficient, not necessarily the same as in (5).

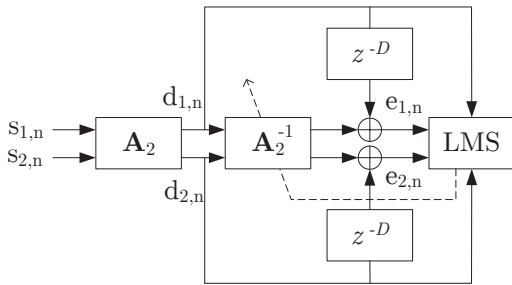


Fig. 3 Identifying the secondary path

### 3 Sound field calculation

#### 3.1 Acoustic wave propagation

To account for all important acoustic phenomena in the operating range of ANC systems below 2 kHz, the simulation is based on the numerical solution of the acoustic wave equation. Our implementation is based on the second order acoustic wave equation (8)

$$\frac{\partial^2 p}{\partial t^2} = c^2 \nabla^2 p, \quad (8)$$

where  $p$  is the acoustic pressure,  $t$  denotes time and  $c$  is the speed of sound [18]. The equation makes a compact statement about the acoustic field and relies on a single spatial variable, hence it is easily implementable and requires the least amount of memory of all other formulations. It is often used in room acoustics, and is fairly accurate given the following conditions: the field under investigation should have a constant, homogeneous air density, time dependent changes in particle velocity should be small compared to the static values and particle velocity should be substantially smaller than the sound velocity [13].

The formulation accounts for wave diffraction and interference phenomena that are important at the low frequencies at

which ANC systems operate. It should be noted that a simulation based on this equation cannot describe situations with extreme sound pressure levels (>120 dB) or when convection is present.

#### 3.2 Reflective surfaces

Ordinary materials used in construction or in furniture will reflect the most part of incoming sound waves. A fundamental measure of this material property is *absorption* usually denoted by  $\alpha$ , i.e. a  $0 < \alpha < 1$  real-valued coefficient defined as the ratio of absorbed and incident energy of the pressure wave.

Most materials have an absorption of less than 10% in the frequency range below 2 kHz, which means that loud noises can be audible even after hundreds of successive reflections [19].

Another related quantity of importance is the reflection factor or *reflectance* denoted by  $R$ , i.e. a  $0 < |R| < 1$  complex-valued coefficient defined as the ratio of the reflected and incoming wave pressures.

While room acoustic measurements favor absorption for its simplicity, the mathematical formulae related to the wave equation contain reflectance since it includes phase information. Their relation is given by

$$\alpha = 1 - |R|^2. \quad (9)$$

The reflection model implemented in our simulation assumes that environments are described by large, smooth surfaces. In this case an abrupt change in material properties will result in a *specular reflection* of the incoming wave. A more detailed model would also account for *diffuse reflections* that are caused by small surface unevenness or gradual changes in wave impedance, which scatters waves. However at low frequencies where ANC is concerned, waves are largely unaffected by the roughness of ordinary building materials due to their large wavelength. Acoustic diffusers designed specifically for this purpose can affect sounds in the range of ANC as well, thus implementing a model for such surfaces is considered for future research.

For a wave travelling in the positive  $\xi$  direction the relation between the acoustic pressure  $p$ , the particle velocity  $u$  of the incoming wave and the boundary impedance  $Z_\omega$  is given by

$$p = Z_\omega u_\xi. \quad (10)$$

Substituting this expression into the conservation of momentum and eliminating particle velocity yields the *boundary condition* for a boundary in the positive  $\xi$  direction.

$$\frac{\partial p}{\partial t} = -c \Xi_\omega \frac{\partial p}{\partial \xi}, \quad (11)$$

where  $\Xi_\omega = \frac{Z_\omega}{\rho c}$  is the normalized wall impedance with  $\rho$  being the density of air and  $c$  the speed of sound. The angle dependent reflectance of such boundary is given by



$$R(\theta) = \frac{\Xi_{\omega} \cos \theta - 1}{\Xi_{\omega} \cos \theta + 1} \quad (12)$$

[13]. While the general solution for waves in a room is given by (8) our simulation will apply variations of (11) at the surface points of objects. The solution inside the objects has no relevance to the simulation.

### 3.3 Scenarios in an unbounded medium

Edge cases of room acoustics can include situations where waves travel for large distances without being reflected at all. Such situations can occur in the presence of an open window, being outdoors or inside of an anechoic chamber. The lack of reflections means that transfer functions can be characterized by two parameters: an attenuation and a delay. The performance of ANC systems improves drastically under such conditions.

In these situations the solution of the wave equation in a certain region of space is affected by the fact that no reflections occur outside of it. For practical reasons the computations must be limited to a bounded volume around our region of interest. This limitation means that the solution of the wave equation must be truncated in a way that causes no reflections.

While such *non-reflecting boundary conditions* exist in one and two dimensions, their existence in three dimensions seems unlikely and consequently scientific interest has shifted towards the research of *absorbing boundary regions*.

The *perfectly matched layer* (PML) technique [20] involves the creation of such an absorbing region around a bounded domain of interest.

The technique is based on *complex coordinate stretching*. The solutions and equations are analytically continued along complex spatial variables, e.g. a complex  $\xi$  contour. In the region of interest this contour is the same real variable as before, but outside the region it takes a growing imaginary part that causes oscillating waves to decay exponentially.

Let  $\mathbf{v}(\underline{\xi}, t)$  be a solution of the wave equation in infinite space, where  $\underline{\xi} = [\xi, \eta, \zeta]^T$ . Assuming that the space far from the region of interest is homogeneous, linear and time invariant, this solution can be given as a superposition of plane waves.

$$\mathbf{v}(\underline{\xi}, t) = \sum_{\mathbf{k}, \omega} \mathbf{V}_{\mathbf{k}, \omega} e^{j(\mathbf{k} \cdot \underline{\xi} - \omega t)}, \quad (13)$$

where  $\mathbf{V}_{\mathbf{k}, \omega}$  are the amplitudes corresponding to the solutions with the wavevector  $\mathbf{k}$  and angular frequency  $\omega$  [21]. These radiating solutions can be decomposed into functions of the form

$$\mathbf{V}(\eta, \zeta) e^{j(k\xi - \omega t)}. \quad (14)$$

Since (14) is an analytical function of  $\xi$  it can be analytically continued for complex  $\xi$  values. In order to achieve the intended decaying behavior we will evaluate it for a  $\xi$  contour, which remains unchanged in the region of interest but gains an

imaginary part outside. This imaginary component will result in an exponentially decaying solution, where  $\Im(\xi) > 0$  [21].

Note, that the solutions are unchanged where  $\Im(\xi) = 0$ , thus the absorbing region causes no reflections, hence it is perfectly matched to the region of interest.

While the basic idea is simple, deriving such coordinate transformations in the time domain is far from trivial and requires a sophisticated mathematical approach. There is a wide variety of PML formulations and the best time-domain implementation is still under debate [22,23]. We have chosen a particular formulation that suits our purposes, which is introduced in Section 4.4.

## 4 Implementation

The simulation program consists of two major components: an active noise controller and an acoustic simulator. The noise controller as described in Section 2.2 is presented as two signal processing structures in the discrete time domain. The software implementation of the FIR filtering described by the DSP structures, and of the explicit update Eqs. (5) and (7) are trivial.

The following subsections will present explicit, discrete update equations for the acoustic field values that are similarly straightforward to implement.

### 4.1 The finite-difference time domain method

The FDTD solves differential equations in an iterative manner from one timestep to the next, while substituting function derivatives with finite differences. The finite differences are calculated from discretized function values along a uniform grid both in space and in time.

Our implementation is based on the numerical solution of (8), which is composed of the second-order derivatives of pressure with respect to time and space. Finite-difference approximations for the second derivative can be chosen arbitrarily, and they will result in FDTD schemes of differing complexity and accuracy.

A finite difference formula for  $f''(x)$  is

$$f''(x) \approx \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}, \quad (15)$$

which is the central difference of  $f''(x)$ , with the least amount of terms [10]. It results in a scheme that is simple to implement and easy to adapt to other modeling techniques, such as those related to reflective surfaces. Moreover central differences yield better accuracy than one-sided approximations.

The accuracy of the formula is described by the *local truncation error* that is the lowest order term of the difference of the Taylor series and the approximated function derivative. In the case of (15) the lowest order term is proportional to  $h^2$ , hence it is a second-order accurate finite difference approximation [13].

## 4.2 Solving the acoustic wave equation

Expanding the nabla expression in (8) we get

$$\frac{\partial^2 p}{\partial t^2} = c^2 \left( \frac{\partial^2 p}{\partial \xi^2} + \frac{\partial^2 p}{\partial \eta^2} + \frac{\partial^2 p}{\partial \zeta^2} \right). \quad (16)$$

To obtain a numerical solution we substitute the partial derivatives with central differences given by (15).

$$\begin{aligned} & \frac{p_{\xi,\eta,\zeta}^{n+1} - 2p_{\xi,\eta,\zeta}^n + p_{\xi,\eta,\zeta}^{n-1}}{\tau^2} \\ &= c^2 \cdot \left( p_{\xi+\epsilon,\eta,\zeta}^n + p_{\xi-\epsilon,\eta,\zeta}^n + p_{\xi,\eta+\epsilon,\zeta}^n + p_{\xi,\eta-\epsilon,\zeta}^n \right. \\ & \quad \left. + p_{\xi,\eta,\zeta+\epsilon}^n + p_{\xi,\eta,\zeta-\epsilon}^n - 6p_{\xi,\eta,\zeta}^n \right) / \epsilon^2 \end{aligned} \quad (17)$$

where  $\epsilon$  and  $\tau$  are the spatial and temporal step sizes. Rearranging terms and substituting a single coefficient in place of the physical variables yields

$$\begin{aligned} & p_{\xi,\eta,\zeta}^{n+1} \\ &= \lambda^2 \left( p_{\xi+\epsilon,\eta,\zeta}^n + p_{\xi-\epsilon,\eta,\zeta}^n + p_{\xi,\eta+\epsilon,\zeta}^n \right. \\ & \quad \left. + p_{\xi,\eta-\epsilon,\zeta}^n + p_{\xi,\eta,\zeta+\epsilon}^n + p_{\xi,\eta,\zeta-\epsilon}^n \right) \\ & \quad + 2(1 - 3\lambda^2) p_{\xi,\eta,\zeta}^n - p_{\xi,\eta,\zeta}^{n-1}, \end{aligned} \quad (18)$$

where  $\lambda = c\tau/\epsilon$  is the Courant number [13].

The term on the left side of the equation is the pressure value at the next timestep  $n + 1$ , while terms on the right are pressure values at the current and previous timesteps  $n$  and  $n - 1$ . In other words a finite-difference time domain approximation estimates a spatial value at the next iteration as the linear combination of previous values in its vicinity. The spatial structure of this vicinity is characteristic of the FDTD formulation and is referred to as a finite difference scheme. The structure described by (18) is called the *simple leapfrog scheme* (SLF).

The particular form of the stencil is determined by the finite difference formulae substituted for the partial derivatives. Different schemes have differing accuracy and stability requirements. In general schemes using a larger number of function values to estimate the next iteration will be more accurate but also require more computational work. There are more efficient schemes than the SLF i.e. schemes that yield higher modelling accuracy relative to their computational cost, but the simplicity of its implementation still makes it appealing.

FDTD schemes aren't inherently stable i.e. the finite difference approximations of physical processes may only be stable within a certain range of parameters. The SLF scheme described above is stable if and only if  $\lambda^2 \leq 1/3$  [13].

The accuracy of the scheme is also dependent on its parameters. A significant metric of FDTD schemes in this regard is the *dispersion error*. It relates to the fact that the wave propagation speed of a scheme is frequency and direction dependent and will differ from the theoretical value  $c$ . The frequency dependent *isotropy error* is the relative value of the *dispersion error* in the

worst case direction [13]. It is a strictly increasing function of frequency, thus it can be limited by observing an  $f_c$  upper cutoff frequency in the simulation. A good rule of thumb for this limit is keeping the wave propagation speed error under 2%.

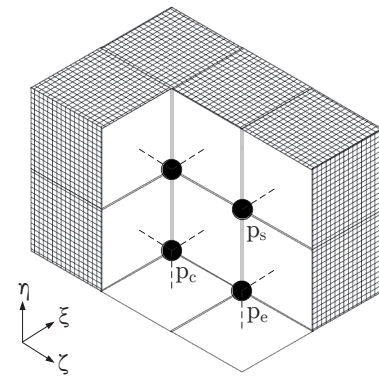
Keeping in mind the criteria of stability and isotropy error the following default parameters have been chosen for the simulation software.

**Table 1** Summary of FDTD parameters

$c$	340 m/s
$\epsilon$	0.02 m
$1/\tau = f_s$	32 kHz
$f_c$	2 kHz

## 4.3 Locally reacting surfaces

The simulation software is using a frequency dependent locally reacting surfaces (LRS) implementation presented by [13]. Figure 4 shows the various cases where Eq. (18) has to be altered due to an impeding material surface.



**Fig. 4** Simulation points next to a reflecting body

In the case of  $p_s$  the point in the positive  $\zeta$  direction falls inside an obstructing body. Hence the equation corresponding to this point has to be modified and the  $p_{\xi+\epsilon,\eta,\zeta}^n$  term has to be replaced with an expression relating to this surface point's acoustic impedance. Such an expression can be obtained by substituting the derivatives in (11) with centered differences, which yields

$$\frac{p_{\xi,\eta,\zeta}^{n+1} - p_{\xi,\eta,\zeta}^{n-1}}{2\tau} = -c\Xi_\omega \left( \frac{p_{\xi+\epsilon,\eta,\zeta}^n - p_{\xi-\epsilon,\eta,\zeta}^n}{2\epsilon} \right) \quad (19)$$

[13]. This discrete equation is the basis for all LRS related modifications of the wave equation.

The acoustic impedance of materials is frequency dependent, which has to be taken into account for accurate auralization simulations, but has little importance in the low frequency range of ANC systems. Nevertheless [13] presents a frequency

dependent LRS method, which has been included in our implementation.

In this case the wall impedance is realized by a digital impedance filter (DIF), in the IIR form

$$\Xi_{\omega}(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}}. \quad (20)$$

In order to correspond to a physically correct model the DIF should be calculated from a reflection filter model  $R(z)$  using

$$\Xi_{\omega}(z) = \frac{1 + R(z)}{1 - R(z)}. \quad (21)$$

In order to iteratively solve (19) and (20) the intermediate value  $g^n$  has to be introduced. The impedance filter's input  $g_{in}^n$  and output  $g_{out}^n$  are expressed in relation to this intermediate value [13].

The complete iterative solution for a field value with a positive  $\zeta$  boundary is given by Eqs. (22)-(25).

$$g^n = \sum_{i=1}^N (b_i g_{in}^{n-i} - a_i g_{out}^{n-i}) \quad (22)$$

$$\begin{aligned} p_{\xi,\eta,\zeta}^{n+1} = & \left[ \lambda^2 \cdot (2p_{\xi-\epsilon,\eta,\zeta}^n + p_{\xi,\eta+\epsilon,\zeta}^n + p_{\xi,\eta-\epsilon,\zeta}^n + p_{\xi,\eta,\zeta+\epsilon}^n \right. \\ & \left. + p_{\xi,\eta,\zeta-\epsilon}^n) + 2(1 - 3\lambda^2) p_{\xi,\eta,\zeta}^n + \frac{\lambda^2}{b_0 g_n} \right. \\ & \left. + \lambda \frac{a_0}{b_0 - 1} p_{\xi,\eta,\zeta}^{n-1} \right] / \left( 1 + \lambda \frac{a_0}{b_0} \right) \end{aligned} \quad (23)$$

$$g_{in}^n = \frac{a_0}{\lambda b_0} (p_{\xi,\eta,\zeta}^{n+1} - p_{\xi,\eta,\zeta}^{n-1}) - \frac{g^n}{b_0} \quad (24)$$

$$g_{out}^n = \frac{1}{a_0} (b_0 g_{in}^n + g^n). \quad (25)$$

Equation (19) describes how the pressure relates to the local impedance in case of a wave travelling in the positive  $\zeta$  direction. Such equations can be similarly constructed for waves travelling in the five other axial directions.

The point  $p_e$  shown in Fig. 4 is an edge point, which means that two of its neighboring points fall within the volume of a reflecting body, while  $p_c$  is a corner point with three neighbors inside an obstruction. In these cases two and three points must be substituted with appropriate expressions in (18) respectively, and each substitution requires a corresponding impedance filter. This means that in case of a corner point three impedance filters must be operated simultaneously and each one is defined by the material their respective neighboring points fall within.

#### 4.4 Perfectly matched layer

The PML technique implemented in the software is presented by [22]. It describes a formulation directly for the wave equation in its second-order form. This and the fact that it only requires four auxiliary variables in three dimensions makes it a particularly efficient implementation.

When the PML function is activated a 32-cell thick outer layer of the simulation region is treated specially by solving a PML modified variant of (18) instead of the original.

The technique requires four auxiliary variables initialized to zero at the beginning of the simulation. The  $\Phi_{1...3}$  auxiliary variables describe a vector field with the subscripts 1 – 3 corresponding to the three spatial components. The  $\Phi_{1...3}$  variables are staggered in space i.e. while the pressure field is stored for points  $(\xi, \eta, \zeta)$ , values corresponding to  $\Phi$  are stored for  $(\xi' = \xi + \frac{\epsilon}{2}, \eta' = \eta + \frac{\epsilon}{2}, \zeta' = \zeta + \frac{\epsilon}{2})$ . The  $\psi$  auxiliary variable is similarly staggered in time but not in space. The auxiliary variables have no physical meaning, they are merely required for the solution of the corresponding differential equations in the time domain.

The PML's behavior can be influenced by its thickness and the damping factor  $\sigma$ . For perfect impedance matching this factor must be zero at the boundary and a smoothly increasing  $\sigma_R(r)$  function of the distance from the simulation region.

$$\sigma_R(r) = \bar{\sigma} \left( \frac{|k_R - a_R|}{L_R} - \frac{\sin\left(\frac{2\pi|k_R - a_R|}{L_R}\right)}{2\pi} \right), \quad (26)$$

for  $a_i \leq |k_R| \leq a_R + L_R$  and 0 otherwise, where  $k_R$  is the spatial index along axis  $R$ ,  $a_R$  is the size of the computation region [22],  $\bar{\sigma} = 4000$  is an adjustable constant and  $L_R = 32$  is the thickness of the PML.

Equations (27)-(29) summarize the calculations in the order of execution.

$$\psi_{\xi,\eta,\zeta}^{n+\frac{1}{2}} = \tau \cdot p_{\xi,\eta,\zeta}^n + \psi_{\xi,\eta,\zeta}^{n-\frac{1}{2}} \quad (27)$$

$$\begin{aligned} p_{\xi,\eta,\zeta}^{n+1} = & \left[ \frac{c^2}{\epsilon^2} (p_{\xi+\epsilon,\eta,\zeta}^n + p_{\xi-\epsilon,\eta,\zeta}^n + p_{\xi,\eta+\epsilon,\zeta}^n + p_{\xi,\eta-\epsilon,\zeta}^n \right. \\ & \left. + p_{\xi,\eta,\zeta+\epsilon}^n + p_{\xi,\eta,\zeta-\epsilon}^n - 6p_{\xi,\eta,\zeta}^n) \right. \\ & \left. + \frac{1}{\epsilon} (\bar{\Phi}_{1\xi+\epsilon,\eta,\zeta}^n - \bar{\Phi}_{1\xi,\eta,\zeta}^n + \bar{\Phi}_{2\xi,\eta+\epsilon,\zeta}^n - \bar{\Phi}_{2\xi,\eta,\zeta}^n \right. \\ & \left. + \bar{\Phi}_{3\xi,\eta,\zeta+\epsilon}^n - \bar{\Phi}_{3\xi,\eta,\zeta}^n) - \frac{\sigma_{1\xi}\sigma_{2\eta}\sigma_{3\zeta}}{2} \right. \\ & \left. \cdot (\psi_{\xi,\eta,\zeta}^{n+1} + \psi_{\xi,\eta,\zeta}^{n-1}) - \frac{1}{\tau^2} (p_{\xi,\eta,\zeta}^{n-1} - 2p_{\xi,\eta,\zeta}^n) \right. \\ & \left. - (\sigma_{1\xi}\sigma_{2\eta} + \sigma_{2\eta}\sigma_{3\zeta} + \sigma_{3\zeta}\sigma_{1\xi}) p_{\xi,\eta,\zeta}^n \right. \\ & \left. + (\sigma_{1\xi} + \sigma_{2\eta} + \sigma_{3\zeta}) p_{\xi,\eta,\zeta}^{n-1} \right] \\ & / \left( \frac{1}{\tau^2} + \frac{\sigma_{1\xi} + \sigma_{2\eta} + \sigma_{3\zeta}}{2\tau} \right) \end{aligned} \quad (28)$$

$$\Phi_{1\xi,\eta,\zeta}^{n+1} = \left[ \frac{\bar{\sigma}_{1\xi'} \Phi_{1\xi,\eta,\zeta}^n}{2} + (\bar{\sigma}_{2\eta'} + \bar{\sigma}_{3\zeta'} - \bar{\sigma}_{1\xi'}) D_{\eta} P_{\xi',\eta',\zeta'}^{n+\frac{1}{2}} + (\bar{\sigma}_{2\eta'} \cdot \bar{\sigma}_{3\zeta'}) D_{\xi} \psi_{\xi',\eta',\zeta'}^{n+\frac{1}{2}} + \frac{1}{\tau} \Phi_{1\xi,\eta,\zeta}^n \right] \left/ \left( \frac{\bar{\sigma}_{1\xi'}}{\tau} \right) \right., \quad (29)$$

where

$$D_{\xi}^h p_{\xi',\eta',\zeta'}^{n+\frac{1}{2}} = \frac{1}{2\epsilon} (\bar{p}_{\xi+\epsilon,\eta',\zeta'}^{n+1} - \bar{p}_{\xi,\eta',\zeta'}^{n+1} - \bar{p}_{\xi+\epsilon,\eta',\zeta'}^n - \bar{p}_{\xi,\eta',\zeta'}^n) \quad (30)$$

$$D_{\xi}^h \psi_{\xi',\eta',\zeta'}^{n+\frac{1}{2}} = \frac{\bar{\psi}_{\xi+\epsilon,\eta',\zeta'}^{n+\frac{1}{2}} - \bar{\psi}_{\xi,\eta',\zeta'}^{n+\frac{1}{2}}}{\epsilon}. \quad (31)$$

and the cell averages for  $p$  and  $\psi$  are defined as

$$\bar{p}_{\xi,\eta',\zeta'}^n = \frac{1}{4} (p_{\xi,\eta,\zeta}^{n+} + p_{\xi,\eta,\zeta}^n + p_{\xi,\eta+\epsilon,\zeta}^n + p_{\xi,\eta+\epsilon,\zeta}^n) \quad (32)$$

$$\bar{\psi}_{\xi,\eta',\zeta'}^n = \frac{1}{4} (\psi_{\xi,\eta,\zeta}^{n+\frac{1}{2}} + \psi_{\xi,\eta,\zeta+\epsilon}^{n+\frac{1}{2}} + \psi_{\xi,\eta+\epsilon,\zeta}^{n+\frac{1}{2}} + \psi_{\xi,\eta+\epsilon,\zeta+\epsilon}^{n+\frac{1}{2}}) \quad (33)$$

The updated equations for  $\Phi_{2\xi,\eta,\zeta}^{n+1}$ ,  $\Phi_{3\xi,\eta,\zeta}^{n+1}$  are analogous to (29).

#### 4.5 Simulation sequence

In order to test the efficiency of the ANC system in a certain environment, both the field simulation and the control algorithm should be run. The straightforward solution is a simultaneous operation as it can be seen in Fig. 5a. All the necessary parameters describing the acoustic environment and the controller are initialized, then field simulation starts. The controller samples the pressure field at the microphone positions and excites the acoustic system at the loudspeaker positions. After the transient phase the sound levels can be evaluated at each position of the enclosure resulting in the suppression map.

According to the experience of the authors, the transient phase of an ANC system can be considerably long, it can reach even tens of seconds. As field simulation is a resource-consuming task, in most cases it cannot be calculated in real-time. Thus ANC simulation according to Fig. 5a would take a long time.

To alleviate this problem, our simulation sequence resembles the steps depicted in Fig. 5b. Having the input parameters first the discrete time transfer functions  $\mathbf{R}(z)$ ,  $\mathbf{A}_1(z)$  and  $\mathbf{A}_2(z)$  are determined. This requires a field simulation with a length of the reverberation time of the room. Then the controller's operation is simulated yielding the adaptive filters' transfer function

matrix  $\mathbf{W}(z)$ . As this stage requires only the operation of a discrete time network, it can be evaluated very quickly. This stage is run until the steady state value of  $\mathbf{W}(z)$  is obtained. In the third step, the field simulation is run again, with the previously calculated  $\mathbf{W}(z)$  incorporated. The transfer functions  $\mathbf{R}(z)$ ,  $\mathbf{A}_1(z)$  and  $\mathbf{A}_2(z)$  are once again realized by the field simulation. The required length of the simulation is not greater than the reverberation time, again. At the end the suppression map can be calculated similarly to the first solution.

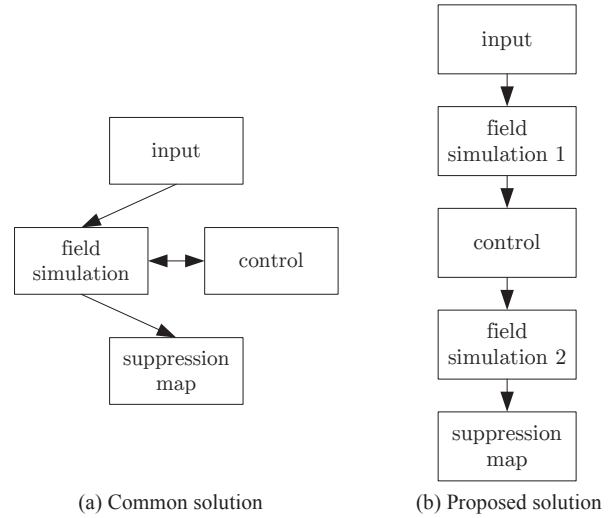


Fig. 5 Procedure diagram of the simulation sequence

To sum up, the resource-consuming calculations should be done for twice the reverberation time, which is much less than that of the first version. In the following the details of our simulation sequence are introduced.

Before the simulation would be executed, the user's inputs are interpreted. The inputs comprise of an environmental description and an interactive module description, in our case an ANC system. The environmental description consists of a 3D polygon model of the acoustic environment, a set of digital impedance filters describing the reflective properties of the model surfaces and a list of acoustic source positions and corresponding wave data. The ANC system is described by a list of positions for actuators, error and reference microphones respectively, as well as internal parameters such as the order of modelling filters, their initial values and the value of the step coefficients. The inputs are characterized by floating point parameters in meters. Starting the core simulation life-cycle all input parameters are discretized according to the FDTD grid resolution, which is 0.02 m and 32 kHz by default.

As stated before, the main simulation consists of three distinguishable stages: field simulation 1, controller operation and field simulation 2. The last stage is a full solution of our original problem, where the acoustic environment modelling and ANC simulation is carried out in parallel.

The stage 'field simulation 1' creates a lumped parameter model of the acoustic environment, which consists of



the impulse responses connecting all points of interest. This involves the estimation of an impulse response from every source position to every sensing position. Source positions include noise or actuator positions, and sensing positions include error and reference microphones, as well as user specified positions where sound levels must be monitored. In order to obtain an accurate model, the impulse responses must be at least as long as the reverberation time of the environment. They are determined by the impulse method, where a bandlimited impulse is emitted from a source location, and the resulting pressure field is recorded at every microphone location. The impulse responses modeling the  $\mathbf{R}(z)$  and  $\mathbf{A}_1(z)$  transfer functions of Fig. 1a can be estimated by emitting a pulse from the points corresponding to  $q_1(z)$  then  $q_2(z)$ . Similarly estimating  $\mathbf{A}_2(z)$  of Fig. 1b is done by emitting the pulses from the points corresponding to  $y_1(z)$  then  $y_2(z)$ . For a system of  $K$  sources and a reverberation time of  $T$  the acoustic wave equation has to be solved for a period of  $K \cdot T$ . For a single noise source and an actuator, with a reverberation time less than 2 seconds, this requires the field simulation of 4 seconds irrespective of the number of error and reference microphones, since the wave equation is already solved at all points of our environment simultaneously.

The stage ‘controller operation’ uses the lumped parameter model to estimate how the ANC module filters will adapt as a result of operating in the given environment. The module is capable of suppressing wideband signals, by iteratively estimating the necessary paths shown in Fig. 1a and 1b. During this iterative estimation the error of the system exponentially approaches an optimal level, until the system reaches a steady state. We are mainly interested in the noise cancelling efficiency of the steady state. Reaching this may even take minutes in practical systems. In this stage acoustic signal propagation and internal module algorithms are simulated for this period. The input of this stage is the wave data describing the noise signals we want to cancel. In Figure 1a these signals are denoted by  $q_1(z)$  and  $q_2(z)$ . The preprocessor uses the appropriate impulse responses from the lumped parameter model to obtain the ANC system’s  $s_1(z)$  and  $s_2(z)$  inputs. The simulated ANC system’s  $\mathbf{W}(z)$  adaptive filter is iteratively approximated using (5), while the system’s  $e_1(z)$  and  $e_2(z)$  inputs are being calculated by using the impulse responses corresponding to the  $\mathbf{A}_1(z)$  and  $\mathbf{A}_2(z)$  transfer functions.

During the stage ‘field simulation 2’ the numerical solution of the wave equation and the ANC module’s algorithms are evaluated simultaneously. Instead of using impulse responses, the inputs of the ANC system are directly determined by the values of the pressure field. Similarly the outputs of the module are superposed directly on the pressure field. The goal of this stage is to create a three-dimensional model of the acoustic suppression. The steady state adaptive filter coefficients are carried over from the stage ‘controller operation’. The consequence of

this, is that the ANC system will immediately operate at optimal efficiency, and the complete system reaches a steady state after the reverberation time. Using the field values of the steady state the noise canceling efficiency can be determined at all points of the acoustic environment.

## 5 Software features and technology

The software enables the user to describe three-dimensional scenes in which sound sources and microphones can be placed. During the simulation the sources will emit sound described by wave data, and the microphones will pick up the observed pressure level, also stored as wave data. The user can instantiate interactive DSP modules with microphone inputs and speaker outputs. For our purposes we have implemented an ELMS noise canceling module that can be configured for an arbitrary number of inputs and outputs. Such modules can be implemented in OpenCL language – a subset of C99 – and added to the simulation suite without having to recompile the program.

Scenes can be edited graphically using the Wings 3D polygon modeler. Each object in the scene has to be assigned to a material that describes how the object will interact with acoustic waves. The materials exhibit a locally acting frequency dependent absorption described by IIR filter coefficients.

The various functions can be accessed through a Python wrapper library. The wrapper offers simple functions for loading models and describing simulation parameters. Wave inputs, outputs and field values can be accessed as numpy arrays, which provides close integration with the mathematical toolset of the NumPy/SciPy scientific – signal processing stack.

An online visualization capability provides a live view of the acoustic field values as the simulation progresses in time. This allows for quick, qualitative detection of problems that require the adjustment of simulation parameters. The visualization is a networked MJPEG stream that can be monitored remotely, while the computer running the simulation need not have graphical capabilities.

The tasks pertaining to the simulation of wave propagation are computationally very demanding. In order to keep simulation times reasonably low present day hardware resources must be utilized by low level, highly parallelized solutions. For this reason the high-performance components of the software have been implemented in C++ and OpenCL with a high-level Python wrapper for handling user interaction. Our implementation was targeting mainly GPU devices, because the FDTD method is particularly well suited for the efficient utilization of such devices, while dedicated graphics cards have about 10 times higher raw computational performance and memory bandwidth than similarly priced CPUs.

OpenCL is a language targeting high-performance computations. It focuses on the description of data-parallel algorithms i.e. computations where the same operations are executed on multiple inputs. It is very similar to Nvidia’s proprietary Cuda

solution, but is supported by a wide range of manufacturers and devices. Computations described in OpenCL can be efficiently carried out on CPUs and GPUs as well. In OpenCL the algorithms can be described in the form of scalar calculations as opposed to vectorized commands such as AVX intrinsics. The calculations take place in *kernels* that provide the executables for lightweight threads. During the simulation thousands of such threads are created and executed by the GPU's own scheduler.

On an i7 3770 CPU even a fairly naive solution in OpenCL is yielding about 100 times better performance than a single threaded Matlab code, while running the simulations on a GPU provides a further 3 – 15 times improvement.

The performance metrics shown in Table 2 were obtained during the simulation of a 5 x 7.5 x 2.8 m room.

**Table 2** CPU – GPU performance comparison

Device	iteration time [ms]	computational cost
		$\left[ \frac{s}{m^3 \cdot \text{real} - \text{life s}} \right]$
Intel i7 3770	51.5	10.79
AMD R9 290	3.3	0.69

The implementation is memory bandwidth bound, which is clearly demonstrated by the performance comparison as the R9 290 has a 15 times higher bandwidth of 320 GB/s. Running this simulation on the GPU took 105 seconds for each elapsed real-life second. At this pace, using our tool is many times preferable to carrying out real life measurements.

## 6 Simulation results

The tool has been evaluated in a wide range of situations including those in an anechoic chamber. Many of these tests were initially carried out as real life measurements and now served as a point of reference, to which our simulation results could be compared. Out of these we have chosen a series of measurements and simulations, which investigate the effectiveness of using multiple reference microphones in a reflective environment disturbed by stochastic noise. Due to their complexity such investigations benefit greatly from the ease of simulation instead of real life instrumentation.

In addition to this specifically selected measurement series, we also present the results of two simulations for the purpose of verification.

### 6.1 Software verification

The software's functional correctness has been established by running simple simulations with theoretically predictable results.

#### 6.1.1 Modes of a rectangular room

We have created a simulation with a rectangular room of  $3 \times 5 \times 7$  m long sides. The wall of the room had a frequency independent reflectance of 99%. During the simulation we have emitted a bandlimited impulse of 2 kHz from one of the corners and picked up the resulting impulse response with a virtual microphone. Observing the Fourier-transform of the impulse response, the modal frequencies could be easily identified as distinct peaks. Table 3 shows the frequencies of the first 10 modes.

**Table 3** Modal frequencies of a rectangular room

Mode [i, j, k]	Simulated [Hz]	Theoretical [Hz]
0,0,1	34	34
1,0,0	57	58
1,0,1	66	67
0,0,2	68	69
0,1,0	87	88
1,0,2	88	90
0,1,1	93	95
0,0,3	101	103
1,1,0	103	105
1,1,1	108	111

Even above 108 Hz all modes appear as expected, however the modal frequencies are lower than the theoretical values. It is apparent that at higher frequencies the simulated modes appear with an increasing error. This is most likely due to the isotropy error which causes waves of higher frequencies to travel slower than their theoretical speed. As explained in section 2 we are observing an upper cutoff frequency of 2 kHz in order to limit this error to 2%, so errors of this magnitude were expected and are acceptable.

#### 6.1.2 Reverberation time

A compact way of describing the acoustic properties of a room is by specifying how quickly sounds decay in the room. This measure is the *reverberation time*. It is denoted by  $RT_{60}$ , and it signifies the time it takes for the energy of a wave to be attenuated by 60 dB.

The relation between  $RT_{60}$  and a room's geometrical and material absorption properties have long been of particular interest. A practically useful equation derived in an empirical fashion is given by (34) called the Sabine equation:

$$RT_{60} = \frac{24 \ln 10}{c_{20}} \frac{V}{Sa} \approx 0.1611 \frac{s}{m} \frac{V}{Sa}, \quad (34)$$

where  $c_{20}$  is the speed of sound at 20°C,  $V$  is the volume of the room,  $S$  is the total surface area of the room and  $a$  is the average absorption coefficient of room surfaces [24].

Equation (35) is the Norris–Eyring formula, which is a more accurate approximation of the reverberation time, but both equations are widely used in room acoustic applications to this day.

$$RT_{60} = \frac{0.161 V}{-S \ln(1-a)}, \quad (35)$$

where the variables  $S$ ,  $V$  and  $a$  denote the same quantities as in (34) [25].

We have created a simulation series to assess our software’s accuracy in creating realistic reverberation times. We have investigated how the simulations behave as the room’s total surface area and its absorption coefficient changes.

The simulations were ran with an empty rectangular room of sides  $L_\xi$ ,  $L_\eta$  and  $L_\zeta$  and a single absorption coefficient  $\alpha$ . During the simulations we have used a 2 kHz bandlimited noise source and a virtual microphone. After simulating 10 seconds we have identified the impulse response between the source and the microphone and determined the time it takes for the power of the impulse response to decrease by 60 dB.

Table 4 shows the simulation results in comparison with the theoretical values.

**Table 4**  $RT_{60}$  in a rectangular room

$\alpha$ [%]	$L_\xi, L_\eta, L_\zeta$ [m]	Simulated [s]	Sabine [s]	N–E [s]
10	$3 \times 4 \times 5$	0.97	1.06	1.01
14	$3 \times 4 \times 5$	0.70	0.72	0.67
19	$3 \times 4 \times 5$	0.53	0.55	0.49
19	$4 \times 5 \times 6$	0.62	0.69	0.63
19	$5 \times 6 \times 7$	0.68	0.83	0.76

The simulated results conform to the theoretical values regarding both the room size and the absorption value.

## 6.2 ANC performance investigation

Acoustic noise canceling is a resource constrained problem. Using an ADSP 21262 signal processor and a well optimized implementation of the ELMS algorithm we can operate adaptive filters with a cumulative tap number of 16000 with a sampling frequency of 8 kHz.

Given the aforementioned complexity of acoustic fields and the length of impulse responses in room acoustic environments we had to make a decision regarding the allocation of these taps. On one hand we can use long adaptive filters and have a more accurate approximation of the transfer paths, but on the other hand, lowering adaptive filter order allows us to process more reference inputs and model more transfer paths.

The presented measurements and simulations investigate the tradeoff between larger filter order and more reference microphones.

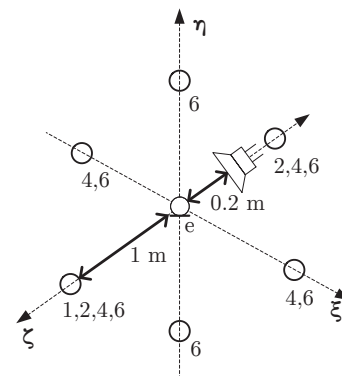
The measurements and simulations took place in an 8 by 11 by 2.8 m room with only a handful of chairs and tables in it, which produced an audibly more reverberant surrounding than ordinary living quarters. The room was located in a multi storey building and had a concrete floor and ceiling and plasterboard walls. On one end it had a wall-to-wall glass window, on the other a large door and metal whiteboard. Along one side it had large metal containers.

Since our investigations concern the low frequency range under 2 kHz the various surfaces in the room were modeled with a single frequency independent absorption value. A list of these values is given by Table 5.

**Table 5** Absorption of materials in the room model

material	absorption [%]
concrete	5
wood	8
wall	8
glass	3
door	8
metal	3

The noise canceling region has been located in the middle of the room 1.5 m above the floor. A single noise source has been placed at the far end of the room, and the reference microphones were placed at a one meter radius around the error microphone. The placement of the test equipment is shown in Fig. 6.



**Fig. 6** ANC component placement

The actuating speaker is located 0.2 m from the error microphone denoted by  $e$ . We have worked with setups using 1 – 6 reference microphones. These are shown with circles in the figure, the numbers under them signify the measurements in which they were included. Figure 7 shows the noise canceling equipment’s position inside the room, where  $n$  denotes the noise source emitting a 1 kHz bandlimited white noise.

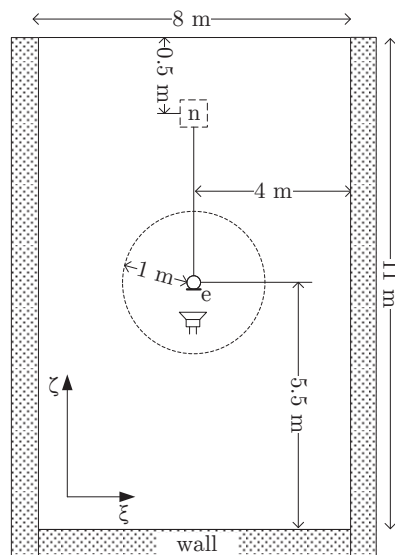


Fig. 7 ANC equipment placement

The noise canceling system was running the ELMS algorithm with one error microphone, one actuator and 1 – 6 reference microphones. The filter corresponding to the secondary path had 460 taps, a sufficiently large number given the close proximity of the actuator and error microphone. The primary filter lengths were chosen so that they would add up to a total of about 16000. This means that measurements with more reference microphones were carried out with lower order adaptive filters. The sampling frequency was 8 kHz.

At the beginning of each measurement the primary filter coefficients were set to 0. The noise source was emitting a continuous 1 kHz white noise. After registering the initial noise reading of the error microphone of about 80 dBC, we have enabled the system's ANC mode. We have established a fixed settling time of 180 seconds, which was enough for the system to reach near maximum suppression. After this interval has elapsed we have registered the suppression value.

The parameters of these measurements were exactly replicated in the software simulations.

Table 6 presents the achieved acoustic suppression values for the measurements and simulations.

Observing the results it is clear that favoring more reference microphones instead of large adaptive filters is very beneficial in room acoustic conditions.

Even as the adaptive filter length goes below the room's reverberation time of roughly 1 s the noise canceling performance keeps improving with the inclusion of additional reference microphones.

Results obtained with the simulation software follow the measured values very closely. According to this using software simulation seems like a feasible alternative when real life measurements would be cumbersome, and even if they aren't, it can be useful for independent verification purposes.

Table 6 Noise suppression in large room

Nr. ref. mic.	Adaptive filter length	Sim. [dBC]	Meas. [dBC]
1	16000	3.0	3.3
2	8000	8.4	6.6
4	4000	13.5	12.5
6	2400	17.6	15.7

## 7 Conclusion

We have presented a simulation tool for evaluating the behavior of active noise control in various acoustic environments. Our framework integrates the two components required for ANC simulations, i.e. sound field calculation and noise control. Field calculation is based on the high-performance implementation of scalable, robust FDTD techniques, while the discrete time controller is realized in a usual way. The novelty of our approach is that the simulation is performed in three stages. First the program determines the discrete transfer function models of the acoustic paths between specific points of interest. In the next stage the controller's operation is evaluated without field simulation. Having the steady state parameters of the adaptive controller, the field simulation is re-run for the reverberation time providing the sound pressure map. Thus the total simulation time is much less than it would be in the case of the simultaneous simulation of the controller and the acoustic field. Sound field calculation is implemented on modern GPGPU devices.

The accuracy of the field calculation has been verified by simulating a simple reverberating enclosure. The results were compared to those obtained through real life ANC experiments. ANC has been tested in a slightly damped room, where the controlled region has been surrounded by the reference microphones. The controller achieved 3 to 16 dB suppression depending on the number of reference microphones, which has been predicted fairly well by the simulations.

Based on the first experiences the program can be efficiently used for ANC performance evaluation. In the near future, the database of the software should be extended by the parameters of different materials. Further research is needed on determining the necessary level of detail for models, that represent frequency dependent, reflective surfaces, in order to obtain sufficiently accurate simulations. Another interesting question is whether the phenomenon of diffuse reflection has to be modeled for ANC.

## References

- [1] Kajikawa, Y. "Recent advances on active noise control: open issues and innovative applications." *APSIPA Transactions on Signal and Information Processing*. 1. pp. 1-21. 2012. DOI: [10.1017/ATSIP.2012.4](https://doi.org/10.1017/ATSIP.2012.4)



- [2] Kuo, S. M., Morgan, D. R. "Active noise control: a tutorial review." In: *Proceedings of the IEEE*. 87 (6). pp. 943-973. 1999. DOI: [10.1109/5.763310](https://doi.org/10.1109/5.763310)
- [3] Kidner, M. R. F. "Active noise control: a review in the context of the 'cube of difficulty'." *Acoustics Australia*. 34 (2). pp. 65-69. 2006.
- [4] Elliott, S. J., Nelson, P. A. "Active noise control." *IEEE Signal Processing Magazine*. 10 (4). pp. 12-35. 1993. DOI: [10.1109/79.248551](https://doi.org/10.1109/79.248551)
- [5] Elliot, S. J., Stothers, I., Nelson, P. A. "A multiple error LMS algorithm and its application to the active control of sound and vibration." In: *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 35 (10). pp. 1423-1434. 1987. DOI: [10.1109/TASSP.1987.1165044](https://doi.org/10.1109/TASSP.1987.1165044)
- [6] Tu, Y., Fuller, C. R. "Multiple reference feedforward active noise control, part I-II." *Journal of Sound and Vibration*. 233 (5). pp. 745-759., pp. 761-774.  
DOI (Part I): [10.1006/jsvi.1998.2077](https://doi.org/10.1006/jsvi.1998.2077)  
DOI (Part II): [10.1006/jsvi.1998.2076](https://doi.org/10.1006/jsvi.1998.2076)
- [7] Computer aided room acoustics. 2015. [Online.] Available from: <http://www.cara.de/ENU/index.html>. [Accessed: 30th July 2015]
- [8] Enhanced acoustic simulation for engineers. 2015. [Online]. Available from: <http://ease.afmg.eu/>. [Accessed: 30th July 2015]
- [9] Odeon room acoustics software. 2015. [Online]. Available from: <http://www.odeon.dk/>. [Accessed: 30th July 2015]
- [10] Taflove, A., Hagness, S. C. "*Computational electrodynamics: The finite-difference time-domain method*." 3rd ed. Artec House, 2005.
- [11] Hughes, T. J. R. "*The finite element method: Linear static and dynamic finite element analysis*." Dover, 2000.
- [12] Wrobel, L. C. "*The boundary element method: Applications in thermo-uids and acoustics*." Vol. 1. Wiley, 2002.
- [13] Kowalczyk, K. "Boundary and medium modelling using compact finite difference schemes in simulations of room acoustics for audio and architectural design applications." PhD thesis. Queen's University Belfast, 2008.
- [14] van Mourik, J., Murphy, D. "Geometric and Wave-Based Acoustic Modelling Using Blender." In: *Audio Engineering Society Conference: 49th International Conference: Audio for Games*. 2013. URL: <http://www.aes.org/e-lib/browse.cfm?elib=16659>
- [15] Kelly, I. J., Boland, F. M. "Detecting Arrivals in Room Impulse Responses With Dynamic Time Warping." *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*. 22 (7). pp. 1139-1147. 2014. DOI: [10.1109/TASLP.2014.2321472](https://doi.org/10.1109/TASLP.2014.2321472)
- [16] Widrow, B., Walach, E. "*Adaptive inverse control*." Prentice Hall, Inc., 1996.
- [17] Elliot, S. J. "Optimal controllers and adaptive controllers for multichannel feedforward control of stochastic disturbances." *IEEE Transactions on Signal Processing*. 48 (4). pp. 1053-1060. 2000. DOI: [10.1109/78.827539](https://doi.org/10.1109/78.827539)
- [18] Morse, P. M., Ingard, K. U. "*Theoretical acoustics*." McGraw-Hill, 1968.
- [19] Physikalisch-Technische Bundesanstalt, alpha-databank. [Online] Available from: <http://www.ptb.de/cms/en/fachabteilungen/abt1/fb-16/ag-163/room-acoustics/absorption-grade-data-collection.html>. [Accessed: 5th April 2015]
- [20] Berenger, J.-P. "A perfectly matched layer for the absorption of electromagnetic waves." *Journal of Computational Physics*. 114 (2). pp. 185-200. 1994. DOI: [10.1006/jcph.1994.1159](https://doi.org/10.1006/jcph.1994.1159)
- [21] Johnson, S. G. "Notes on perfectly matched layers (PMLs)." Massachusetts Institute of Technology, 2010. [Online]. Available form: <http://math.mit.edu/~stevenj/18.369/pml.pdf> [Accessed: 5th April 2015]
- [22] Grote, M. J., Sim, I. "Efficient PML for the wave equation." 2010. Available from: <http://arxiv.org/abs/1001.0319v1>. [Accessed: 30th July 2015]
- [23] Hastings, F. D., Schneider, J. B., Broschat, S. L. "Application of the perfectly matched layer (PML) absorbing boundary condition to elastic wave propagation." *The Journal of the Acoustical Society of America* 100 (5). pp. 3061-3069. 1996. DOI: [10.1121/1.417118](https://doi.org/10.1121/1.417118)
- [24] Sabine, W. C. "*Collected papers on acoustics*." Cambridge: Harvard University Press, 1922.
- [25] Eyring, C. F. "Reverberation time in 'dead' rooms." *The Journal of the Acoustical Society of America*. 1 (2A). p. 168. 1930. DOI: [10.1121/1.1901884](https://doi.org/10.1121/1.1901884)